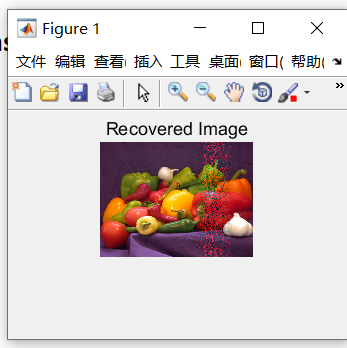
**Answer for class exercise**

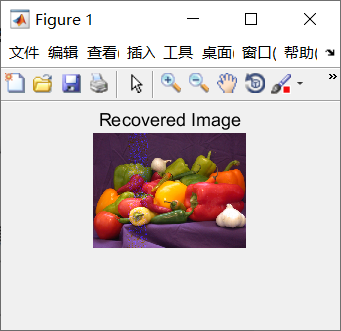
**11911303 吉辰卿**

**恢复图像：**

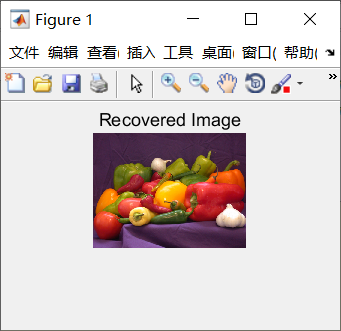
**RxWaveform1.mat**

****

**RxWaveform2.mat**

****

**RxWaveform3.mat**

****

**Codes:**

function[rxSymbols,rxEncodedBits,outLen]=subframeProc(enb,sf,rxGrid,frame,LFrame,Lsf,cec, rxSymbols)

rxSymbols = []; txSymbols = [];

hcd = comm.ConstellationDiagram('Title','Equalized PDSCH Symbols',...

'ShowReferenceConstellation',false);

enb.NSubframe = sf;

rxsf = rxGrid(:,frame\*LFrame+sf\*Lsf+(1:Lsf),:);

[hestsf,nestsf] = lteDLChannelEstimate(enb,cec,rxsf); %

pcfichIndices = ltePCFICHIndices(enb);

[pcfichRx,pcfichHest] = lteExtractResources(pcfichIndices,rxsf,hestsf);

[cfiBits,recsym] = ltePCFICHDecode(enb,pcfichRx,pcfichHest,nestsf);

enb.CFI = lteCFIDecode(cfiBits);

[pdschIndices,pdschIndicesInfo] = ltePDSCHIndices(enb, enb.PDSCH, enb.PDSCH.PRBSet);

[pdschRx, pdschHest] = lteExtractResources(pdschIndices, rxsf, hestsf);

[rxEncodedBits, rxEncodedSymb] = ltePDSCHDecode(enb,enb.PDSCH,pdschRx,...pdschHest,nestsf);

rxSymbols = [rxSymbols; rxEncodedSymb{:}];

outLen = enb.PDSCH.TrBlkSizes(enb.NSubframe+1);

[decbits{sf+1}, blkcrc(sf+1)] = lteDLSCHDecode(enb,enb.PDSCH,... outLen, rxEncodedBits);

txRecode = lteDLSCH(enb,enb.PDSCH,pdschIndicesInfo.G,decbits{sf+1});

txRemod = ltePDSCH(enb, enb.PDSCH, txRecode);

[~,refSymbols] = ltePDSCHDecode(enb, enb.PDSCH, txRemod);

txSymbols = [txSymbols; refSymbols{:}];

release(hcd);

step(hcd,rxEncodedSymb{:});

end